

C L A I M S

- 09748450 122600
1. Method for running in parallel at least one parallel method (28) associated with a sequential caller program means, the method comprising the step of:

issuing (220) a dedicated parallelization call to a parallel program managing means (18,22,26) comprising all control information needed to allow for running said parallel method (28) in parallel.
 2. The method according to claim 1 further comprising the step of

serializing (210) input arguments for said subprogram means,

running (230, 240) said parallel method (28) in parallel on a different machine yielding a result ,

returning (250, 260, 270) said result to the caller program,

deserializing (280) the result.
 3. The method according to claim 1 further comprising the step of generating said parallel method (28) with a script program means which in turn is arranged to invoke a stream editor in order to fill a template means with the code or the name of the method (28) to be computed in parallel.
 - a 4. The method according to ~~the preceding claim~~ ^{Claim 3}, further comprising the step of

automatically generating the instantiation of said template means.
 - a 5. The method according to ~~the preceding claim~~ ^{Claim 4} in which a script is used for generating parallel subprograms.
 6. The method according to claim 1 in which said dedicated

a

- Claim 6

- Sub A178

- [illegible]

- 23a

- a

- a

script means for generating parallel subprograms.

Claim 12

- a 13. The library according to ~~the preceding claim~~ which provides the prerequisites to generate user library functions that make parallelism transparent to a caller of said user library functions.
14. User library generated by means of the library according to claim 12.
- a 15. The library according to claim 12 ~~or claim 14~~ which is a dynamic link library.
16. A parallel program managing tool comprising program means for returning (250, 260, 270) results from parallel executable subprogram means.

09748450 122600